



the examiner's decision of rejection or  
application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision  
of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-313630 ✓

(P2001-313630A)

(43) 公開日 平成13年11月9日 (2001.11.9)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テマコード* (参考)
H 0 4 L 7/00		H 0 4 L 7/00	Z
G 1 0 L 19/00		H 0 4 J 3/00	M
H 0 4 J 3/00		3/06	Z
3/06		G 1 0 L 9/18	M
H 0 4 N 7/24		H 0 4 N 7/13	Z

審査請求 未請求 請求項の数 7 O L (全 9 頁)

(21) 出願番号 特願2001-88447(P2001-88447)

(22) 出願日 平成13年3月26日 (2001.3.26)

(31) 優先権主張番号 00400854.6

(32) 優先日 平成12年3月29日 (2000.3.29)

(33) 優先権主張国 欧州特許庁 (E P)

(71) 出願人 595033034

ドイチェ トムソン-ブランド ゲーエム  
ベーク

Deutsche Thomson-Br  
andt GmbH

ドイツ連邦共和国 デー78048 ヴィリ  
ンゲン-シュヴェニンゲン ヘルマン-シ  
ュヴェアー-シュトラッセ 3

(74) 代理人 100070150

弁理士 伊東 忠彦 (外1名)

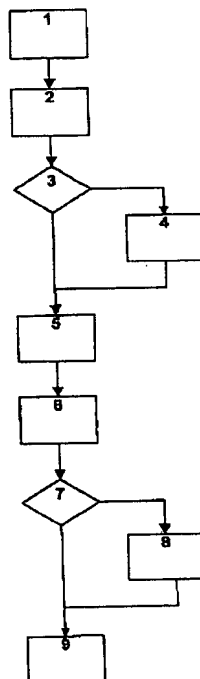
最終頁に続く

(54) 【発明の名称】 オーディオ又はビデオデータ符号化の出力遅延を変化させる方法及び装置

(57) 【要約】

【課題】 操作者の選択に応じて切換可能なビットレートを可能とするためオーディオ符号化器を切換可能に遅延遅延させることを目的とする。

【解決手段】 TV放送のため、関連するビデオ及びオーディオ符号化器を有する符号化システムが使用される。オーディオ及びビデオ遅延は、オーディオ及びビデオストリームを多重化し送信する前に調整される。タイムスタンプ機構によれば、符号化されるべきデータに関連する入力タイムスタンプが発生され、データ遅延定数を用いて入力タイムスタンプから導出される出力タイムスタンプによる出力の前に置き換えられる。入力タイムスタンプは符号化処理の遅延を制御し、出力タイムスタンプは出力時間を示す。切換可能な出力遅延を可能とするためデータ遅延定数は変化される。割当て済み出力タイムスタンプはそのままとされる。未割当て出力タイムスタンプのデータでは出力タイムスタンプは新しい遅延定数を用いて計算される。



## 【特許請求の範囲】

【請求項1】 符号化されるべきオーディオ又はビデオデータと関連付けられ符号化処理の遅延を制御するために使用される入力タイムスタンプが生成され、データ遅延定数を用いることによって上記入力タイムスタンプから出力タイムスタンプが導出され、上記出力タイムスタンプは出力時間を示すために符号化されたデータに割り当てられ、割り当てられたタイムスタンプを伴う符号化されたデータは出力される前にバッファリングされる、オーディオ又はビデオ符号化の出力遅延を変化させる方法であって、

出力遅延を変化させるために上記データ遅延定数が変化されること、既に割り当てられた出力タイムスタンプは変化されないままとされること、及び出力タイムスタンプがまだ割り当てられていないデータについては新しいデータ遅延定数を用いて出力タイムスタンプが計算されることを特徴とする方法。

【請求項2】 遅延が増加されたときに出現する出力タイムスタンプのシーケンス中の間隙の間に埋込みデータ又はゼロデータが送信されるか、又は全くデータが送信されない、請求項1記載の方法。

【請求項3】 遅延が減少されたときに出現する同じ又は重なり合う出力タイムスタンプを有するデータについては、既に経過した出力タイムスタンプを示す後のデータは廃棄される、請求項1又は2記載の方法。

【請求項4】 遅延が減少されるときは、古いデータ遅延定数と新しいデータ遅延定数の間の差に対応する持続時間についての遅延変化要求で始まる更なるデータは出力データバッファには書き込まれず、全ての後続するデータについての出力タイムスタンプの計算には新しいデータ遅延定数が使用される、請求項1乃至3のうちいずれか一項記載の方法。

【請求項5】 オーディオ又はビデオデータの不連続性は、符号化器によって緩和される、請求項1乃至4のうちいずれか一項記載の方法。

【請求項6】 上記不連続性は、間隙又は飛ばされた部分の前にフェードインし、間隙又は飛ばされた部分の後にフェードアウトすることによって緩和される、請求項1乃至5のうちいずれか一項記載の方法。

【請求項7】 請求項1乃至6のうちいずれか一項記載の方法を実行するための装置。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、オーディオ又はビデオデータ符号化の出力遅延を変化させる方法及びオーディオ又はビデオデータ符号化の出力遅延を変化させる装置に関する。

## 【0002】

【従来の技術】 ビデオ符号化器及びオーディオ符号化器を有する符号化システムは、様々な適用、例えばテレビ

ジョン放送のために使用される。この場合、ビデオ符号化器は、例えば選択されたビットレートに依存して最大で1.5秒の可変符号化遅延を有する。従って、消費者製品の復号化器のバッファ制約により、オーディオ及びビデオ遅延は、オーディオストリーム及びビデオストリームを多重化し送信する前に調整されねばならない。

【0003】 タイムスタンプ機構を用いてオーディオ符号化器中の遅延を制御するための基本的な機構は、欧州特許出願第99250009号に記載されている。多伝送路オーディオ符号化器では、少なくとも1つの入力処理段において符号化されるべきオーディオデータのフレームと関連付けられるボードの入力タイムスタンプが発生し、入力タイムスタンプ又は入力タイムスタンプから導出されるタイムスタンプは処理における他の処理段において対応して処理されたフレームデータと関連付けられたままであるが、少なくとも最後の処理段においては出力タイムスタンプによって置き換えられる。これらの各段において、処理の全体の遅延を制御するために処理されるべき現在のフレームデータに関連付けられる対応するタイムスタンプ情報について注目する。

## 【0004】

【発明が解決しようとする課題】 例えば追加的なテレビジョンチャンネル用の空間を作るために、ビデオ符号化器及びオーディオ符号化器について操作者の選択に応じて切換可能なビットレートを許すため、オーディオ符号化器の切換可能な遅延が所望である。しかしながら、欧州特許出願第99250009号は、このような遅延の変化がいかにして扱われるかについて記載していない。

## 【0005】

【課題を解決するための手段】 本発明は、特にビデオ符号化器及びオーディオ符号化器について操作者の選択に応じて切換可能なビットレートのために、オーディオ又はビデオデータ符号化の出力遅延を変化させる方法を提供することを目的とする。この目的は、請求項1に記載の方法によって達成される。

【0006】 本発明は、本発明による方法を用いてオーディオ又はビデオデータの出力遅延を変化させる装置を提供することを更なる目的とする。この目的は、請求項7に記載の装置によって達成される。

【0007】 原理的に、本発明による方法によれば、符号化されるべきオーディオ又はビデオデータと関連付けられ符号化処理の遅延を制御するために使用される入力タイムスタンプが発生される。出力タイムスタンプは、データ遅延定数を用いることによって入力タイムスタンプから導出され、出力時間を示すために符号化されたデータに割り当てられる。割り当てられた出力タイムスタンプを伴う符号化されたデータは、出力される前にバッファリングされ、出力遅延を変化させるためにデータ遅延定数が変化される。既に割り当てられた出力タイムスタンプは変化されていないままである。出力タイムスタ

ンプがまだ割り当てられていないデータについては、新しいデータ遅延定数を用いて出力タイムスタンプが計算される。

【0008】本発明による方法の有利な他の実施例は、各独立項に記載されている。

#### 【0009】

【発明の実施の形態】以下、本発明の実施例について添付の図面を参照して説明する。図1は、出力遅延を変化させる方法を示すフローチャートである。第1の段階1において、オーディオサンプル又はオーディオ係数に対して入力タイムスタンプ情報ITSが関連付けられるか割り当てられる。次に、段階2においてオーディオサンプル又はオーディオ係数が符号化される。段階3において、ユーザ定義の遅延Dが要求されるか否かが検査される。要求される場合は、段階4において、出力タイムスタンプについての処理式 $OTS = ITS + D$ が変化される。次に、段階5において、入力タイムスタンプ情報ITSは出力タイムスタンプOTSと置き換えられ、その後、段階6が行われる。バッファリングは、符号化処理の出力側で行なわれ、なぜならば出力側では必要とされるメモリ量が低いからである。このように、入力データは、一般的にはオーディオ符号化器によって受信された直後に符号化され、符号化した後は特にトランスポートストリームパケットの形状で遅延バッファに入れられる。

【0010】段階9においてデータを出力へ送信する前に、ステップ7においてOTSがチェックされる。OTS中で連続的に間隙が生ずる場合、段階8において埋込みデータ又はゼロデータが挿入される。一方、遅延バッファ中で同一又は重なり合うOTSを有する2つのパケットが見つかった場合も、段階8における特別な処理が必要とされる。1つの可能性は、既に経過した出力時間を示すパケットを廃棄することである。このような場合を扱うには、異なる時間についての遅延要求で始まる出力遅延バッファへ更なるデータを書き込まず、後続の全てのパケットについてOTS計算のための新しい遅延時間を使用することも可能である。最後に、段階9において、データが出力へ送信される。

【0011】以下のデータ処理では、特に段階8において、夫々0.3秒の遅延増加及び遅延減少と、0.5秒の最初の遅延とについて詳細に説明する。

【0012】遅延の増加の場合は、遅延バッファは、現在有効な0.5秒の遅延時間に等しい或るデータ量を含み、このデータの量は、0.8秒まで増加される。これは、遅延バッファからの符号化データの出力が0.3秒間に亘って有効に停止される必要がある一方で、遅延バッファへのデータ入力が継続されることを意味する。

【0013】遅延バッファ中のデータは、直前に既に出力された遅延バッファの継続を表わす既に符号化されたデータである。従って、遅延バッファは、遅延時間につ

いての変化の要求の後に、遅延バッファは、遅延時間変化の時点において遅延バッファ中にある全てのデータが完全に出力されるまで、0.5秒に亘ってデータを送出し続けるよう管理される。次に、遅延バッファからの出力は停止され、これは送信器/復号化器チェーンのいずれかに埋込みゼロデータ、又は、パケットが送信されていない場合を意味する。すると、停止は0.3秒の時間に亘って続き、これはそれに従って遅延バッファの内容を増加させることを必要とする。

【0014】この挙動は、上述のタイムスタンプに基づく遅延制御機構を使用することによって達成される。所与の時間において遅延バッファ内にある全ての出力ブロック、即ちTSパケットもまた、パケットが遅延バッファから送出されその後送信器へ送信されるべき時点を示す出力タイムスタンプが付される。既に遅延バッファ中にあるパケットについては何も変化される必要はなく、それらは生成時に意図されたように出力される。操作者によって直接的又は間接的にオーディオ遅延Dが変化した時点において、出力タイムスタンプ $OTS = ITS + D$ についての処理式が変化され、即ち全てのOTSタイムスタンプは0.3秒ずつ増加される。「古い」遅延で計算された全てのパケットは、出力段によって遅延バッファから順に1つずつ出力される。次に、0.5秒後に、OTSの連続性に間隙があり、即ち、次のパケットは、パケットに遅延変化のない場合よりも0.3秒だけ増加された時間を示すOTSを示す。次に、出力段は、埋込みデータ又はゼロデータを送信するか、又はパケットを送信しない。

【0015】このように、ユーザ即ち復号化器側で聞いている消費者に対する全体的な遅延効果は増加し、遅延変化要求の後は、オーディオプログラムは現在の有効な0.5秒の遅延については通常通り連続し、オーディオプログラムは0.3秒では短く消音し、プログラムは、0.8秒の新しい遅延では通常通り連続する。

【0016】操作者側では、遅延変化要求があると、符号化器に既に入力されている全てのプログラム部分がユーザに対して順次に与えられ、遅延の切換の後に入力された全てのオーディオプログラム部分は、ユーザ側では先行する部分から短い間隔で分離される。

【0017】操作者は、ユーザにとっての不快感を最少とするため、遅延時間変化のためにプログラム間隙又は個々のプログラム部分の間の切換の瞬間を使用しう。

【0018】また、遅延減少のため、遅延バッファは或る量のデータを含む。ここで遅延は0.5秒から0.2秒の遅延まで減少される。この場合、通常は、出力処理を0.3秒間に亘って継続し、遅延バッファへの更なる入力データの書き込みを停止することが必要である。このように、符号化器の入力において入手可能な短い時間のプログラム素材はユーザへは送信されない。原理的には、ユーザ側のオーディオプログラムは連続的であり得

るが、短い断片は信号から切り出される。

【0019】遅延の減少は、同じタイムスタンプに基づく遅延制御によって達成されうる。オーディオ遅延変化要求の直後に、出力タイムスタンプOTSの計算は、OTSが遅延変化のない場合よりも0.3秒早い時点を示すよう変更される。

【0020】遅延バッファへのデータパケットの書き込みが続けられると、遅延バッファ中で同じ又は重なり合うOTSを有する2つのパケットが見つけれられる。遅延バッファ中ではパケットは順序付けられているため、オーディオ遅延変化要求の前の「古い」パケットの出力は、変化要求の時点において遅延バッファの中にあつた全てのデータが出力されるまで、通常通り続けられる。すると、次のパケットは、0.3秒間に亘って既に経過した出力時間OTSを示し、それにより出力ドライバ段はこれらのパケットを廃棄せねばならない。

【0021】このような場合を扱うための他の方法は、0.3秒の差の時間についての変化要求で開始するデータを出力遅延バッファに書き込まず、後続する全てのパケットについてOTS計算のために新しい遅延時間を使用することである。この場合、符号化器の出力段は、多少連続的なOTSシーケンスを見つける。

【0022】このように、ユーザ（復号化器側で聞いている消費者）に対する全体的な遅延効果は、遅延変化要求の後、オーディオプログラムは現在の0.5秒の遅延については通常通り連続し、プログラムは、0.8秒の新しい遅延では通常通り連続するが、0.3秒の遅延差のオーディオプログラムは飛ばされる。

【0023】操作者側では、遅延変化要求があると、符号化器に既に入力されている全てのプログラム部分が順次に、即ち通常通りユーザに対して与えられ、0.3秒の遅延時間差に等しい後続する時間期間についての変化要求の直後に符号化器に供給されるプログラム素材はユーザ側では可聴でなく、この0.3秒の時間期間の後、符号化器に入力される全てのオーディオプログラム部分は通常はユーザ側で可聴である。

【0024】このように、操作者は遅延時間を変化させるためにプログラム内容スイッチを使用することができ、聴いている人に対してはプログラムが全く失われなことを保証するために遅延時間差だけ次のプログラムの開始を単に遅延させる。

【0025】聴いている人に可聴となるオーディオプログラムの全ての不連続性は、符号化器によって任意に緩和され、即ち正しいフェードイン及びフェードアウトが行われうる。遅延の増加とは、間隙の前にフェードアウトし、間隙の後にフェードインすることである。遅延の減少とは、飛ばされた部分の前にフェードアウトし、その後にフェードインすることである。遅延の変化をオーディオプログラムの切替と共に行なう場合、オーディオプログラムはこのスイッチを含むため、これは必要でな

い。

【0026】本発明の方法は、図2に示されるオーディオ符号化器において使用されうる。符号化器は、4つのステレオPCM入力信号PCMA, PCMB, PCMC, 及びPCMDを受信する。例えば、MPEGオーディオデータはフレームに基づくものであり、各フレームは1152のモノサンプル又はステレオサンプルを含む。図2の符号化器オペレーティングシステムは、4つのMPEG伝送路の符号化のために6つのDSP（図示せず）を含みうる。これらのDSPは、図2に示される技術的な機能を含むソフトウェア符号化器を形成する。DSPの適当なタイプは、例えばアナログ装置についてのADSP21060又は21061又は21062である。或いは、図2に示される技術的な機能はハードウェア中に実現されうる。

【0027】6つのDSP又は対応するハードウェア中で実行されるソフトウェアの同期は、夫々バッファが1つ以上の特定のフレームに割り当てられるFIFOバッファを用いて達成される。これは、ある時点において、処理段の中に、現在のフレーム及び先行フレーム、利用可能なバッファの量に依存する数が存在することを意味する。

【0028】幾つかの段の間に、非同期的な読み書き動作を可能とする非同期バッファASBUFが挿入される。他の段の間では、同期バッファBUFで十分である。PCM入力信号PCMA, PCMB, PCMC, 及びPCMDは、夫々非同期バッファを通過して夫々の変換器CONA, CONB, CONC, 及びCONDに送信される。かかる変換器では、符号化されるべきオーディオサンプルの整数-浮動小数表現が行なわれうる。また、符号化器がオーディオサンプルの整数表現を処理することも可能である。かかる変換器では、フレーム中の1つ以上の種類のエネルギーレベル、例えばフレームの全てのサンプルのエネルギー又はフレームのサンプルの平均エネルギーが計算されうる。これらのエネルギー値は、続く心理音響処理に使用されうる。

【0029】更に、かかる変換器では、おそらく適合されている符号化パラメータは、フレームオーディオデータと関連付けられうる。夫々のパラメータ符号化器PENCA, PENCB, PENCC, 及びPENCDでは、元の符号化パラメータは上述のように変換されえ、次に夫々CONA, CONB, CONC, 及びCONDへ供給される。

【0030】非同期バッファを介して、CONA, CONB, CONC, 及びCONDの出力データは、サブバンドフィルタSUBA, SUBB, SUBC, 及びSUBDへ、また、第1の左右伝送路音響心理計算器Psycho1A\_L, Psycho1A\_R, Psycho1B\_L, Psycho1B\_R, Psycho1C\_L, Psycho1C\_R, Psycho1D\_L, 及

び  $Psycho1D\_R$ へ、並列に供給される。サブバンドフィルタは、可能であればFFTを用いて、全体のオーディオスペクトルを周波数帯域へ分周し、周波数帯域又はサブバンド中の係数の最大又はスケールファクタを計算しうる。周波数帯域では、正規化が実行される。サブバンドフィルタは、上述のタイムスタンプ情報を考慮に入れ、できれば対応する上流の非同期バッファから読み出される当該の符号化パラメータを考慮に入れる。

【0031】第1の音響心理計算器は、例えば1024サンプルの長さを有するFFTを実行し、現在のマスク情報を決定する。各第1の音響心理計算器には、サブバンドフィルタにおいて以前に計算された最大値又はスケールファクタ値を評価する第2の音響心理計算器  $Psycho2A\_L$ ,  $Psycho2A\_R$ ,  $Psycho2B\_L$ ,  $Psycho2B\_R$ ,  $Psycho2C\_L$ ,  $Psycho2C\_R$ ,  $Psycho2D\_L$ , 及び  $Psycho2D\_R$  が続く。第1及び第2の音響心理計算器は、上述のタイムスタンプ情報を考慮に入れ、対応する上流の非同期バッファから読み出されるおそ

く関連であろう符号化パラメータを考慮に入れる。

【0032】  $Psycho2A\_L$ ,  $Psycho2A\_R$ ,  $Psycho2B\_L$ ,  $Psycho2B\_R$ ,  $Psycho2C\_L$ ,  $Psycho2C\_R$ ,  $Psycho2D\_L$ , 及び  $Psycho2D\_R$  の出力信号は、割り付けられ量子化されるビットの数を決定するために、ビット割付け器及び量子化器  $Bal/Q/E\_A$ ,  $Bal/Q/E\_B$ ,  $Bal/Q/E\_C$ , 及び  $Bal/Q/E\_D$  において夫々使用され、オーディオデータ係数は、バッファを介して関連付けられるサブ

バンドフィルタから来る。また、第2の音響心理計算器において、第1の音響心理計算器で計算されるもの以外を計算し、それにより第1の音響心理計算器をなくすことも可能である。

【0033】最後に、  $Bal/Q/E\_A$ ,  $Bal/Q/E\_B$ ,  $Bal/Q/E\_C$ ,  $Bal/Q/E\_D$  の出力は、復号化器にステレオ出力信号  $PCM\_Out\_A$ ,  $PCM\_Out\_B$ ,  $PCM\_Out\_C$ ,  $PCM\_Out\_D$  を与える非同期バッファ及び出力インタフェース  $AES-EBU\_A$ ,  $AES-EBU\_B$ ,  $AES-EBU\_C$ , 及び  $AES-EBU\_D$  を夫々通る。出力インタフェースは、IEC958に対応するものであってもよい。

【0034】ビデオ符号化器は、以下の段、即ち、ブロック差分段、DCT（離散コサイン変換）段、量子化、そして帰還ループ内に逆量子化、逆DCT、その出力がブロック差段に入力される動き補償された補間を含み、量子化の出力はできれば最終出力の前に符号化されバッファされたVLC（可変長符号化）であり、バッファ記憶レベルは、符号化アーティファクトが出来る限りマ

クされるよう量子化を制御するために使用される。

【0035】この符号化器では、以下の要素、関係のある全ての入力段及び出力段について固有の値を与える、システム時間を供給するシステム時間基準と、十分に正確な入力タイムスタンプ（ITS）を得るためにシステム時間基準と入力データとを関連付けるハードウェア及び/又はソフトウェア機構と、出力タイムスタンプ（OTS）に従って十分に正確な出力を得るためにシステム時間基準をデータ出力と関連付けるハードウェア及び/又はソフトウェア機構と、が要求される。

【0036】上述の要素は、以下のように使用される。

(a) システムの入力インタフェース毎に、入力データはシステムタイマに関連付けられ、即ち入力タイムスタンプITSは入来するデータと共に得られ、データフレームに割り当てられる。例えば、サンプリングされたオーディオデータブロック又はフレームの第1のサンプルのサンプリング時点のシステム時間がそのために使用される。

(b) 多数の入力がある場合、入力データブロックは入力タイムスタンプによって与えられる時間情報によって伝送路に亘って再び調整されうる。

例1：多伝送路オーディオ入力がかの2伝送路インタフェースに亘って分散されている。

例2：多数のステレオ符号化器のビットストリーム出力は、伝送路間の良く決められた時間の関係で、即ち等しい遅延の可能性で、MPEGTS（トランスポートストリーム）へ多重化される。

(c) 入力タイムスタンプITS及び意図される全体遅延Dから出力データについての出力タイムスタンプOTSが計算される。システムによって1つの入力データブロック当たり1つの出力データブロックが計算される最も簡単な処理の場合は、各出力ブロックの意図される出力時間は  $OTS(n) = ITS(n) + D$  によって表わされ、 $n = 0, 1, 2 \dots$  はデータブロック番号を示す。1つの入力データブロック当たりに幾つかの出力データブロック又は非整数の出力ブロックが発生される場合、各出力ブロックについてのOTSは対応する時間比に従って補間されうる。1つの例は、1152のサンプルが入力されるMPEG符号化器、又は1つ以上のESP（基本ストリームパケット）を有するMPEGPESパケット、及びパケットが188バイトの長さを有するMPEGTSトランスポートストリーム、即ち伝送のために各サンプルフレームにつき3乃至7のTSパケットが必要とされる。

(d) 各出力インタフェースは、処理段によって出力バッファに供給される出力データブロックを、図1について上述されるように現在のシステム時間に関して関連付けられるOTSについて調べる。OTSが、既に経過した時点を示す場合、アプリケーションに依存して、出力データブロックは廃棄されるか、直ぐに出力されうる。

OTSが未来の時点を目指す場合、出力段はその時点が来るまで待ち、待機時間中は何も出力しないか、又は決められた埋込みパターンを出力する。

【0037】実施されるべきI/O処理のために必要とされる異なる組み合わせで使用されうる2つの関連付けられた機構が利用可能である。

【0038】例としての最少ハードウェアシナリオの場合は、システムは、一般的には既に各DSPの一部である単一のハードウェアタイマを、ある種の普通の又は被制御出力ドライバソフトウェアの作動と共に使用する。残る遅延制御は、DSPによって実行される。原理的に、以下の2つのタイマ関数が必要とされる。

・ソフトウェアが実際のシステム時間を尋ねることを可能とする「getTime()」関数。各入力データブロックの開始又は終端を(DMA又はINTに基づき)受信すると、そのブロックについてのITSを得るためにgetTime()関数が使用されうる。

・出力関数は、対応する処理が既に終了している出力ブロックを送信する前に一定の遅延を必要としうる。これは、ポーリング方式で、即ちある種の循環的な作動が利用可能であれば実際のシステム時間対OTSの循環的な制御を行なうこと、又は定義可能な遅延「interruptAfter(DELAY)」又は「threadActivityAfter(DELAY)」関数の後に割り込みを発行する特定の時間に基づく遅延関数によってなされうる。

【0039】ゼロにおける割り込みのある逆カウンタとして動作する単一のハードウェアタイマとDSPの埋込み関数である入力及び出力MDAブロック完了割り込みが使用される。

【0040】単一のハードウェアタイマは、関数「interruptAfter(DELAY)」及び「getTime()」を与えることができ、「getTime()」の関数については、連続的なシステム時間を得るためにタイマにロードされる続く遅延時間は足し合わされ、幾つかの「interruptAfter(DELAY)」関数は並列して実行されうる。

【0041】各DSPがそれ自身のタイマを実施しているが、異なるDSPに亘る要求要求で入力及び出力を分配する多DSPシステムの場合、タイマ同期性の問題がある。この問題は、システム中の全てのDSPに与えられシステム時間を再び同期させるために使用される特別な周期的割り込み信号(例えば長さ10ms)によって解決されうる。カウンタの出力ワードは、例えばiiii.ffffの形式を有するものであってもよく、但しiiiiは整数部であると理解され、ffffは分数部であると理解される。各10msのiiiiは「1」だけインクリメントされる。このイベントは、DSPへ送信され、その中で計数される。DELAYについての最大の管理可能な値は、iiiiのワード長に依存する。それにより、割り込みは再同期の時点を示し、割り込み期間中は1つのマスタDSPから全ての多のDSPへ時間に

ついてのマスタ値iiiiが送信される。この時点においてffffはゼロに設定される。

【0042】上記の例としてのシナリオにおいて上述された遅延制御が最小限のハードウェアのみを必要とする一方で、(より安価でより柔軟性のある)ソフトウェアに殆どの仕事をさせるため、遅延時間の精度が例えば、割り込み待ち時間、最大割り込み不可時間、及び多DSPシステムの場合はバス割当て遅延によって制限されるという欠点がある。達成される精度が十分でない場合、システム時間に対してITSを得る処理及びシステム時間に対するOTSで出力する処理をより正確にするためのハードウェア向上が使用されうる。大体の時間を決めるソフトウェアと正確な時点を決める特別なハードウェアを組み合わせる使用することにより、所望のDSP反応時間(ハードウェアよりも遅い傾向がある)とハードウェアの複雑性(時間期間が長いほど複雑となる傾向がある)との間の折衷案が得られる。

【0043】最後の処理段に関連するバッファにおいて始まり、前処理段に関連するバッファに先行する理論的なオーバーフローは、DSP間のハンドシェイクによって防止される。本発明による遅延を変化させる方法は、単一のDSPシステム及び他の種類のリアルタイム処理にも適用されうる。

【0044】簡単化された適用では、例えばAC-3復号化器では、しばしば用いられるアプローチは、各ブロックについての処理時間がフレーム期間よりも短くなくてはならないというリアルタイム制約での受信の後に単一のデータフレームを計算するものである。このアプローチは、例えば最終的には幾つかのDSPへ分散される幾つかの続く処理段へ分割する場合を含むより分散された処理を含む解法へ拡張されうる。この場合、各処理段はよく決められた「タイムスロット」へ分けられ、各処理段について、処理時間はタイムスロットの長さよりも短くなくてはならない。最初に説明した解法とは異なり、単一の時間制約の代わりに、タイムスロット/処理段毎に1つの時間制約がある。

【0045】MPEGが例えば様々なサンプル周波数及び全体データ速度を可能とするため、符号化器が異なる符号化パラメータで操作することが可能であるという要件がある。

【0046】本発明は、操作者が遅延を変化させたときに、最も短い中断期間が確実とされるという利点を有する。

【0047】本発明は、任意の可変ビットレート符号化器又は可変処理時間を必要とする符号化器について固定であるが可変の符号化遅延を確実とするために使用される。

【0048】本発明は特に、MPEG1, 2, 3のためのMPEG1, 2, 4オーディオ符号化及び復号化、AC-3のためのデジタルビデオ放送DVB、DVD処



理のためのMD及びAAC処理、及びオーディオデータ符号化及び復号化に関するインターネットアプリケーションのために使用されうる。

【図面の簡単な説明】

【図1】出力遅延を変化させる方法を示すフローチャートである。

【図2】本発明による方法を用いた4伝送路オーディオ符号化器を示す機能ブロック図である。

【符号の説明】

1 入力タイムスタンプ情報の割り当て

2 符号化

3 ユーザによって定義される遅延の変化が要求されているか？

4 出力タイムスタンプの処理式を変更

5 入力タイムスタンプを出力タイムスタンプで置き換え

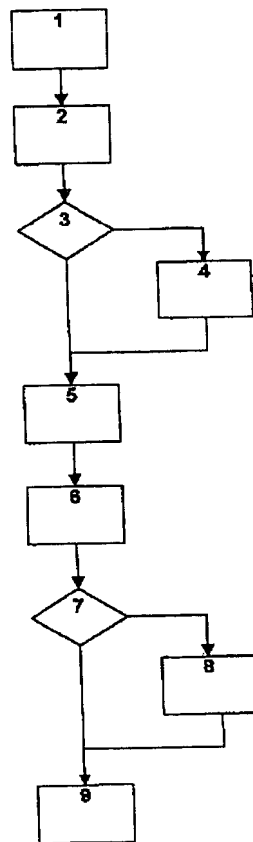
6 バッファリング

7 出力タイムスタンプを検査

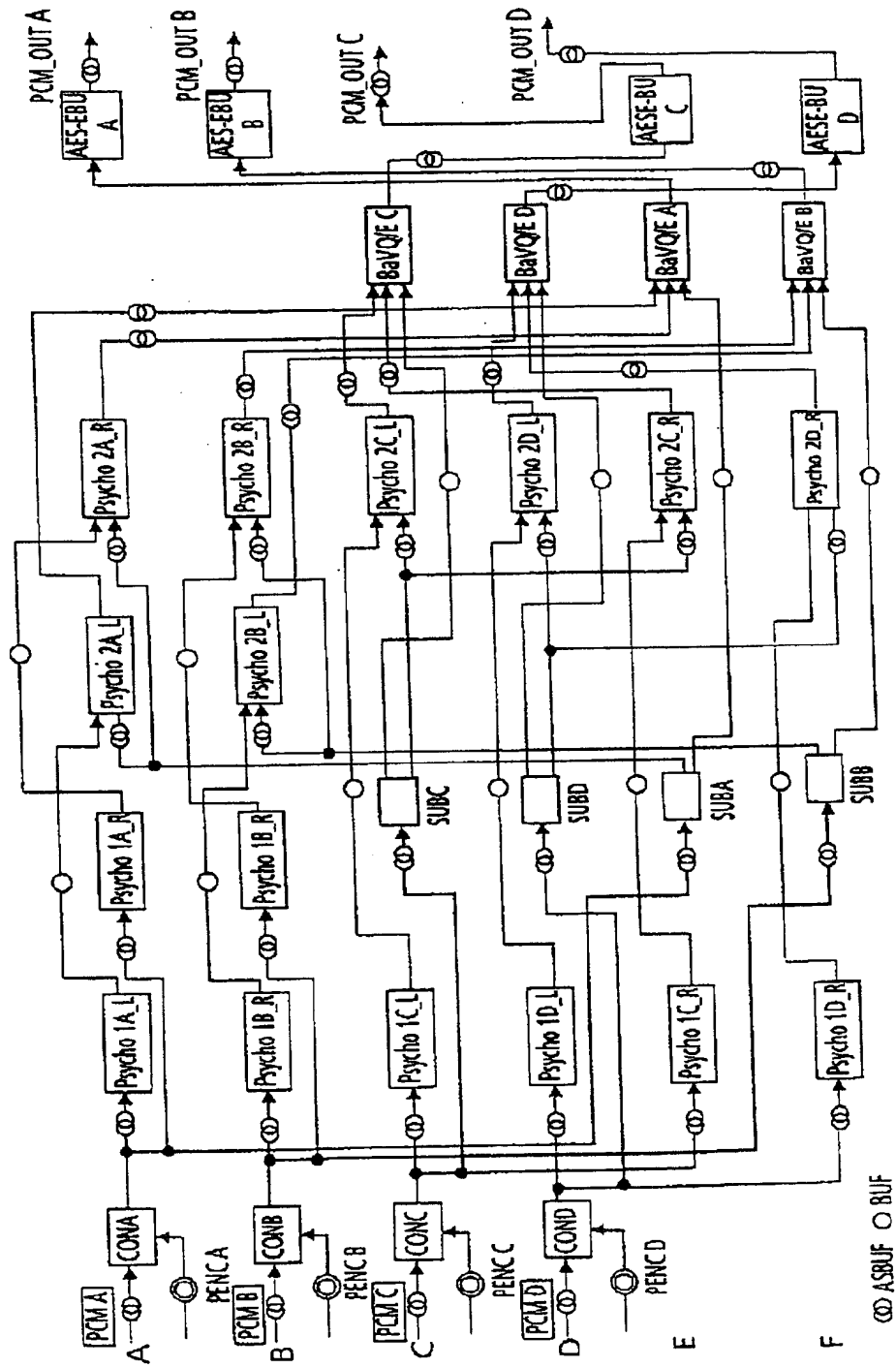
8 埋込みデータの挿入

10 9 データを出力へ送信

【図1】



【図 2】



フロントページの続き

(71)出願人 598147008

トムソン ブロードキャスト システムズ  
THOMSON BROADCAST S  
YSTEMS

フランス国 95800 セルジー・サン・ク  
リストフ リュ・デュ・プティ・アルピ

17

(72)発明者 ウルリヒ シュライバー

ドイツ連邦共和国, 30827 ガルプゼン,  
ライファイゼンシュトラッセ 7

(72)発明者 ステファン シューケ

フランス国, 35690 アシニエ, リュ・デ  
ュ・シャン・ムリエ 4